**Automatic Data Visualization**

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

It enables decision makers to see analytics presented visually, so they can grasp difficult concepts or identify new patterns. With interactive visualization, you can take the concept a step further by using technology to drill down into charts and graphs for more detail, interactively changing what data you see and how it’s processed.

 If you’ve ever stared at a massive spreadsheet of data and couldn’t see a trend, you know how much more effective a visualization can be. The concept of using pictures to understand data has been around for centuries, from maps and graphs in the 17th century to the invention of the pie chart in the early 1800s.

A picture is worth a thousand words. A good visualization is worth a terabyte of data. Nowadays, the ability to create good visualizations has shifted from a nice-to-have skill to a must-have skill for all data analysts to help managers make business decisions, where data is the primary force behind for its high volume and overwhelming velocity. Despite the overwhelming choices of interactive data visualization tools for experts, non-experts have poor choices for effective visualization recommendation systems such that everyone can easily create great visualizations.

The current data visualization tools have allowed users to create good visualizations, only if the users know their data well. Ideally, the users need tools to automatically recommend visualizations, so they can simply pick the ones they like. This is hard, if not impossible, since among numerous issues, no consensus has emerged to quantify the goodness of a visualization that captures human perception.

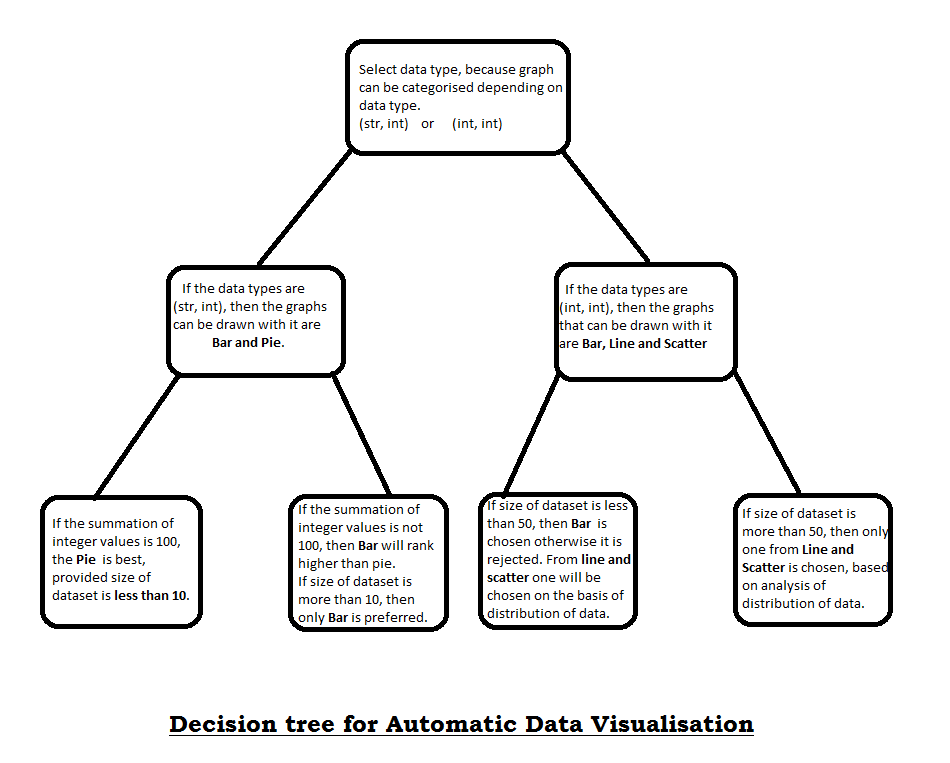
Technically speaking, “interesting” charts can be defined from three angles:

**(1) Deviation-based:** a chart that is dramatically different from the other charts.

**(2) Similarity-based:** charts that show similar trends w.r.t. a given chart.

**(3) Perception-based:** charts that can tell compelling stories, from understanding the data, without being compared with other references.

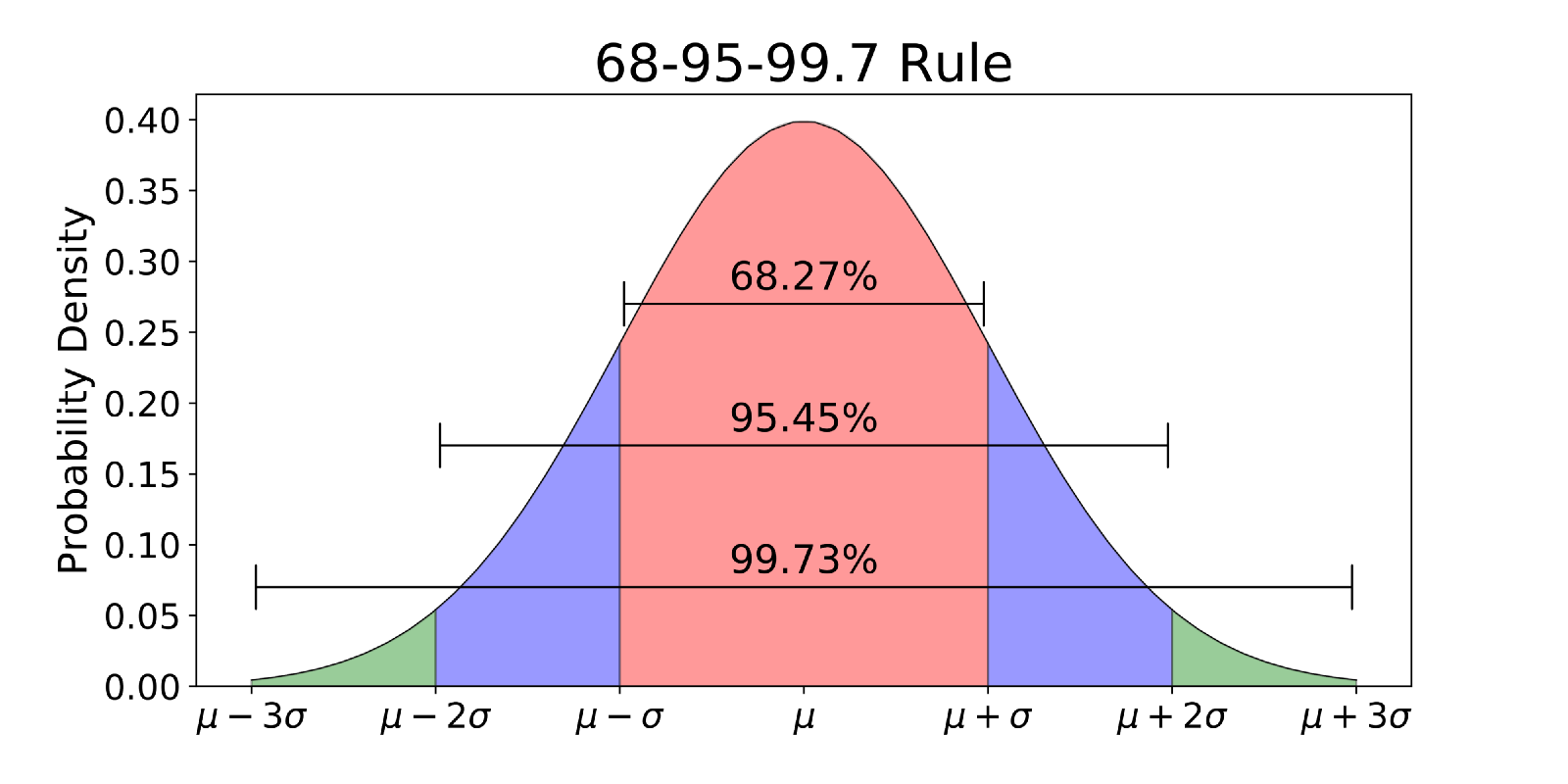
The given below binary classifier is designed by keeping in mind the understanding of human perception. For an example, a bar chart with more than 50 bars is probably a bad bar chart.



The above decision tree can be used to create data for a machine learning model, which can easily classify the labels depending on the type and size of the input data.

Measures of distribution of data:

Normal distribution, also known as the Gaussian distribution, is a [probability distribution](https://www.investopedia.com/terms/p/probabilitydistribution.asp) that is symmetric about the mean, showing that data near the mean are more frequent in occurrence than data far from the mean. In graph form, normal distribution will appear as a bell curve.



**Mean:** The Arithmetic Mean is the average of the numbers: a calculated "central" value of a set of numbers.

To calculate it:

• add up all the numbers,

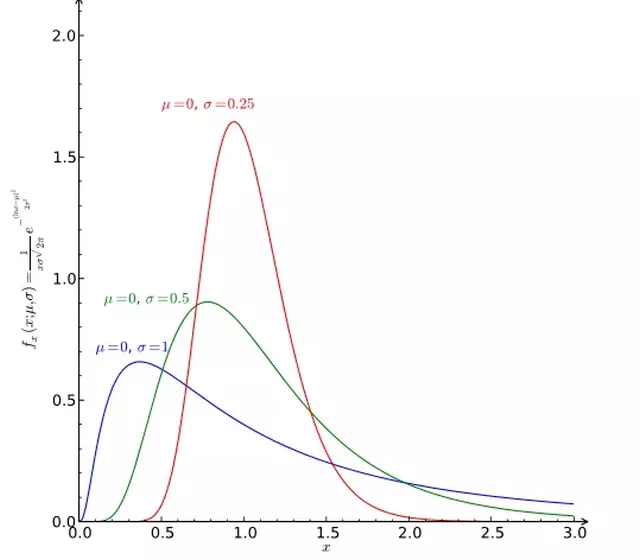
• then divide by how many numbers there are.

**Median:** The Median is the "middle" of a sorted list of numbers.

**Standard deviation:** The standard deviation is a statistic that measures the dispersion of a dataset relative to its mean and is calculated as the square root of the variance. It is calculated as the square root of variance by determining the variation between each data point relative to the mean. If the data points are further from the mean, there is a higher deviation within the data set; thus, the more spread out the data, the higher the standard deviation.

**Skewness:** Skewness, in statistics, is the degree of distortion from the symmetrical bell curve, or [normal distribution](https://www.investopedia.com/terms/n/normaldistribution.asp), in a set of data. Skewness can be negative, positive, zero or undefined. A normal distribution has a skew of zero, while a [lognormal](https://www.investopedia.com/terms/l/log-normal-distribution.asp)distribution, for example, would exhibit some degree of right-skew.

* Skewness, in statistics, is the degree of distortion from the symmetrical bell curve in a probability distribution.
* Distributions can exhibit right (positive) skewness or left (negative) skewness to varying degree.
* Investors note skewness when judging a return distribution because it, like kurtosis, considers the extremes of the data set rather than focusing solely on the average.



The three probability distributions depicted above depict increasing levels of right (or positive) skewness. Distributions can also be left (negative) skewed. Skewness is used along with kurtosis to better judge the likelihood of events falling in the tails of a probability distribution.

**Kurtosis:** Like [skewness](https://www.investopedia.com/terms/s/skewness.asp), kurtosis is a statistical measure that is used to describe the distribution. Whereas skewness differentiates extreme values in one versus the other tail, kurtosis measures extreme values in either tail. Distributions with large kurtosis exhibit tail data exceeding the tails of the normal distribution (e.g., five or more standard deviations from the mean). Distributions with low kurtosis exhibit tail data that are generally less extreme than the tails of the normal distribution.

Kurtosis is a measure of the combined weight of a distribution's tails relative to the center of the distribution. When a set of approximately normal data is graphed via a histogram, it shows a bell peak and most data within + or - three standard deviations of the mean. However, when high kurtosis is present, the tails extend farther than the + or - three standard deviations of the normal bell-curved distribution.

**Z-score:** A Z-score is a numerical measurement used in statistics of a value's relationship to the mean (average) of a group of values, measured in terms of [standard deviations](https://www.investopedia.com/terms/s/standarddeviation.asp) from the mean. If a Z-score is 0, it indicates that the data point's score is identical to the mean score. A Z-score of 1.0 would indicate a value that is one standard deviation from the mean. Z-scores may be positive or negative, with a positive value indicating the score is above the mean and a negative score indicating it is below the mean. It is generally used to compare two entities on different scales.

**Correlation:** Correlation, in the finance and investment industries, is a statistic that measures the degree to which two securities move in relation to each other. Correlations are used in advanced [portfolio management](https://www.investopedia.com/terms/p/portfoliomanagement.asp), computed as the [correlation coefficient](https://www.investopedia.com/terms/c/correlationcoefficient.asp), which has a value that must fall between -1.0 and +1.0.

* Correlation is a statistic that measures the degree to which two variables move in relation to each other.
* In finance, the correlation can measure the movement of a stock with that of a benchmark index, such as the Beta.
* Correlation measures association, but does not tell you if x causes y or vice versa, or if the association is caused by some third (perhaps unseen) factor.

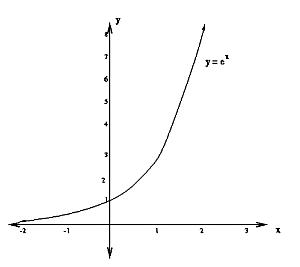
**Binning partitions the numerical or temporal values into different buckets: –**

**Temporal values** are binned by minute, hour, day, week, month, quarter, year, whose data type can be automatically detected based on the attribute values.

**Numerical values** are binned based on consecutive intervals, e.g., bin1[0, 10), bin2[10, 20), ...; or the number of targeted bins, e.g., 10 bins.

• Grouping groups values based on categorical values.

**Scaling** can be calculated on the basis of z-scores of the data, as in exponential data majority data points will have negative z-score.



**Python code for calculating z-score and the ratio of numbers having negative z-scores:**

from scipy import stats

a = np.array([ 2.71, 7.78, 20.58, 54.56, 148.56, 403.42, 1096.78]) //exponential data

score = stats.zscore(a)

count=0

for i in score:

if(i<0):

count=count+1

ratio=count/len(a)